

This Page Is Inserted by IFW Operations  
and is not a part of the Official Record

## **BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning documents *will not* correct images,  
please do not report the images to the  
Image Problem Mailbox.**

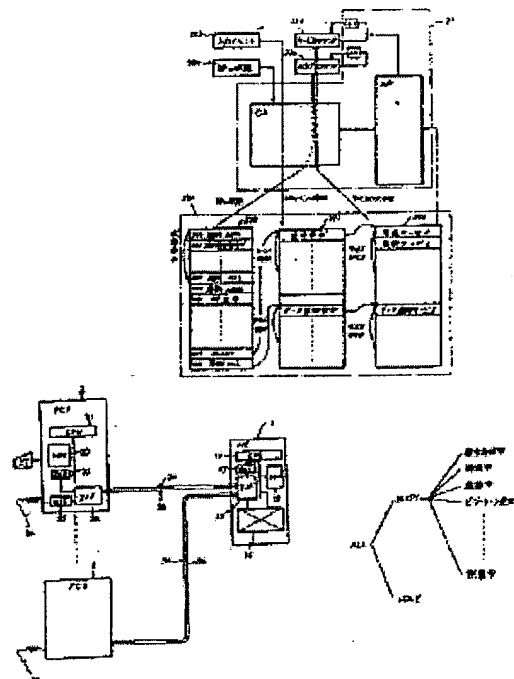
## CALL PROCESSING TASK CONTROL SYSTEM

Patent number: JP60253397  
 Publication date: 1985-12-14  
 Inventor: HAKUTA AKIRA; others: 04  
 Applicant: FUJITSU KK  
 Classification:  
 - international: H04Q3/545; G06F9/46  
 - european:  
 Application number: JP19840109978 19840530  
 Priority number(s):

## Abstract of JP60253397

**PURPOSE:** To prevent the increase of scale in call processing program and also to facilitate the easy change in program such as addition, deletion, etc., by setting the state (e.g., busy, all) of an upper call and deciding a task according to the upper call state, an input event and a task decision table.

**CONSTITUTION:** In case such a task that has the common state of each call to a specific input event type is designated, the state of a high-order call representing the states of calls is set instead of a case where an input event is set at the state of each call. Then a task is decided. For this purpose, a task decision table 22a is provided to a main memory 22 of a high performance terminal 2. Then a call state table 220 of the table 22a stores states 809 all, 907 busy and 908 all of an high-order call in addition to each call state. While an event type-based table 221 stores the input event types (incoming request, data information request, etc.) for execution of a common task. Furthermore a service counter table 222 stores the task names (transfer service, store service, data information service, etc.) which should be executed according to the stored input event types.



Data supplied from the esp@cenet database - Patent Abstracts of Japan

⑨ 日本国特許庁(JP)

⑩ 特許出願公開

⑫ 公開特許公報(A) 昭60-253397

⑪ Int. Cl.

識別記号

庁内整理番号

⑬ 公開 昭和60年(1985)12月14日

H 04 Q 3/545  
G 06 F 9/46

7459-5K  
B-8120-5B

審査請求 未請求 発明の数 1 (全8頁)

⑭ 発明の名称 呼処理タスク制御方式

⑮ 特 願 昭59-109978

⑯ 出 願 昭59(1984)5月30日

⑰ 発 明 者	伯 田 晃	川崎市中原区上小田中1015番地	富士通株式会社内
⑱ 発 明 者	藤 井 義 信	川崎市中原区上小田中1015番地	富士通株式会社内
⑲ 発 明 者	有 高 徳 裕	川崎市中原区上小田中1015番地	富士通株式会社内
⑳ 発 明 者	前 田 潤	川崎市中原区上小田中1015番地	富士通株式会社内
㉑ 発 明 者	森 友 春 男	川崎市中原区上小田中1015番地	富士通株式会社内
㉒ 出 願 人	富士通株式会社	川崎市中原区上小田中1015番地	
㉓ 代 理 人	弁理士 山谷 皓榮		

明 細 書

1. 発明の名称 呼処理タスク制御方式

2. 特許請求の範囲

(1) 呼の状態と入力イベント種別に対応したタスクを実行する呼処理タスク制御方式において、該呼の状態を格納する状態記憶部と、該入力イベント種別を格納するイベント格納部と、該呼の状態と該入力イベント種別とに応じて実行すべきタスクを決定するタスク決定テーブルとを有し、該入力イベント種別に対して共通なタスクを指定する複数の呼の状態の上位の呼の状態を設定するとともに、該タスク決定テーブルに該上位の呼の状態に対応する入力イベント及びタスクを格納しておき、該イベント格納部からの入力イベントと該上位の呼の状態から該タスク決定テーブルによりタスクを決定し、該上位の呼の状態に基づいてタスクが決定できない時に該状態記憶部の呼の状態と該入力イベントから該タスク決定テーブルにより

タスクを決定することを特徴とする呼処理タスク制御方式。

(2) 前記タスク決定テーブルに前記実行すべきタスクに関する優先情報および内容を前もって登録しておき、前記上位又は個々の呼の状態、前記入力イベント格納部から読み出されたイベントおよび前記タスクに関する優先情報をもとに前記タスク決定テーブルから決定されたタスクに基づき所望のタスクを逐次実行することを特徴とする特許請求の範囲第(1)項記載の呼処理タスク制御方式。

3. 発明の詳細な説明

(産業上の利用分野)

本発明は、蓄積プログラム制御により交換サービスを行なうため呼の状態と入力イベントとに応じて対応するサービス(タスク)を実行する呼処理タスク制御方式に関し、特に入力イベントに対するタスクが複数の呼の状態に対し共通であるものにおけるプログラム規模の小規模化を計った呼処理タスク制御方式に関する。

電子交換システムは広く電話交換として利用されており、加入電話器(端末)からの交換サービスを蓄積プログラム制御によつて実行している。このような交換サービスは、呼の状態と入力イベントとによりオペレーティングシステムが実行すべきサービス(タスク)を決定し、そのサービス(アプリケーション)プログラムを実行することによつて所望のサービスを行つている。

例えば、第6図の発信側基本接続サービス状態遷移図で説明すると、idle状態(800)より出発し、call on(呼出要求であり、電話器をフックオンした状態)という入力イベントが生じると、選択信号(相手側の電話番号)受信処理というタスクが選定されたこの処理が実行され、選択信号受信(801)という呼の状態と数字受信完了という入力イベントが生じると、相手側に着信要求(call Req)を発する処理が選定され、これを実行し、接続可応答待(802)という呼の状態となる。以下同様に呼の状態(800~807)と入力イベント(call on, 数字受信完了, 数字受信エラー,

タイムアウト, all off等)とに応じてタスクが決定され、タスクを実行して必要な接続サービスを実行する。第7図は着信側基本サービス状態遷移図であり着信側においても同様に呼の状態(900~910)と入力イベント(call Req, call on等)とによつてタスクが決定され、必要な基本接続サービスが実行される。この様な交換サービスにおいては単なる交換サービスのみならず種々の付加的サービス(例えば転送サービス、蓄積サービス等)が要求されておりその機能向上が求められている。又交換サービスが音声のみならずデータにまで拡大している今日では、データに対するサービス(データ通知サービス等)も要求されている。

例えば第6図、第7図においてビジー(busy)中において、通話相手以外の相手より着信要求があつた場合に転送サービスによつてこれを転送することができる。

〔従来の技術〕

このような転送サービスを行う場合には、呼の状態と入力イベントによつてタスクを決定、実行

するという原則に従えば、第6図のビジー中における各呼の状態801~807に対し、他の相手から着信があるかどうかの入力イベントを設定する必要がある。

第7図においても、ビジー中における各呼の状態(901~910)に対し同様の入力イベントを設定する必要がある。即ち個々の呼の状態から始まるサービスの記述が必要であり、転送サービスの他に種々の付加サービスを実行するには、これら全てのサービスを個々の呼の状態に応じてサービスの記述が必要となる。

〔発明が解決しようとする問題点〕

従つて、呼処理プログラム自体が大規模化し、プログラムのオーバーヘッドが大きくなるという問題が生じていた。また必要なサービスを選択したい要求に対しては呼処理プログラム全体を変更する必要があるという問題もあつた。

〔問題点を解決するための手段〕

本発明は、上述の問題を解決するため、サービスを付加してもプログラム自体のオーバーヘッド

が大となることを防止し、しかもサービスの変更が容易な呼処理タスク制御方式を提供するにある。

このため、本発明は呼の状態と入力イベント種別に対応したタスクを実行する呼処理タスク制御方式において、該呼の状態を格納する状態記憶部と、該入力イベント種別を格納するイベント格納部と、該呼の状態と該入力イベント種別とに応じて実行すべきタスクを決定するタスク決定テーブルとを有し、該入力イベント種別に対して共通なタスクを指定する複数の呼の状態の上位の呼の状態を設定するとともに、該タスク決定テーブルに該上位の呼の状態に対応する入力イベント及びタスクを格納しておき、該イベント格納部からの入力イベントと該上位の呼の状態から該タスク決定テーブルによりタスクを決定し、該上位の呼の状態に基づいてタスクが決定できない時に該状態記憶部の呼の状態と該入力イベントから該タスク決定テーブルによりタスクを決定することを特徴としている。

また、本発明の一実施形態では、前記タスク決

定テーブルに前記実行すべきタスクに関する優先情報および内容を前もって登録しておき、前記上位又は個々の呼の状態、前記入力イベント格納部から読み出されたイベントおよび前記タスクに関する優先情報をもとに前記タスク決定テーブルから決定されたタスクに基づき所望のタスクを逐次実行することを特徴としている。

#### 〔作用〕

本発明では、入力イベントに共通なタスクを指定する複数の呼の状態の上位な呼の状態（例えばビジー、オール）を設定し、この上位な呼の状態と入力イベントからタスク決定テーブルによりタスクを決定しているので、ビジーやオールという上位の呼の状態に対して共通のタスクを決定するのに個々の呼の状態に入力イベントを設定しなくてよく、そのサービス（タスク）を記述しなくてよいので、呼処理プログラムの大規模化を防ぐことができ、また追加、削除等の変更も容易にできる。

#### 〔実施例〕

であり、3aがシグナル・データ用バス、3bは音声用バスであり、高機能端末2とネットワークコントローラ1とを接続するものである。

第2図実施例構成においては、高機能端末2には音声・データの複合端末で構成され、電話器の如く音声通信を行う他に、通常のデータ端末と同様にデータ通信及びデータ処理を行うものであり、例えば、パーソナルコンピュータに音声通信機能を付加したものと考えてよい。そして各呼処理（第6図、第7図）は高機能端末2によつて行われる分散処理形式の通信システムであり、ネットワークコントローラ1は高機能端末2からシグナル・データ用バス3aを通じて与えられる発呼要求、バス接続要求に基いてバスの接続を行うものである。例えば高機能端末PC1が高機能端末PC2に通話するには、高機能端末PC1のキーボード27よりの選択信号を受信するとプロセッサ21は伝送インターフェイス24を介しシグナル・データ用バス3aによつてネットワークコントローラ1に発呼要求を発し、ネットワークコントローラ1はシグナル・データ用バス3aを介し高

以下、本発明を実施例により詳細に説明する。

第2図は本発明の一実施例構成図であり、電子交換システムの全体図であり、図中、1はネットワークコントローラであり、後述する端末相互間にトランスペアレントな通信路を提供するため、ネットワークのスイッチングを行うものであり、スイッチング制御を行うためのプロセッサ11とプロセッサ11のメインメモリ12と、端末との信号リンク用のデータリンク制御部13と、端末との伝送インターフェイス14と、回線交換用のT1段のスイッチング回路15とを有している。2は高機能端末であり、ネットワークコントローラ1に複数台スター（星）状に接続され、通信サービスを制御する機能を有し、ネットワークコントローラ1を介し通信を行うためのものであり、プロセッサ21と、そのメインメモリと、データリンク制御回路23と、伝送インターフェイス24と、音声インターフェイス25と、音声通信用のヘッドセット26（電話器の送受話部）とキーボード27とを有している。3a、3bはバス

機能端末PC2に伝え、PC2では着信分析し、接続可なら接続可信号をシグナル・データ用バス3aを介しネットワークコントローラ1に送り、ネットワークコントローラ1は更にシグナル・データ用バス3aを介し高機能端末PC1にこれを送る。高機能端末PC2では呼出音を発し、同様に高機能端末PC1でもブザー音を発し、高機能端末PC2で応答、即ちフックオンがあると、当該端末PC2から通信可通知をネットワークコントローラ1へシグナル・データ用バス3aを介し送るとともに呼出音を停止する。ネットワークコントローラ1は通信可通知を端末PC1へ与え、端末PC1ではブザー音を停止し、ネットワークコントローラ1にバス接続要求を発する。ネットワークコントローラ1はこれによつて端末PC1とPC2との音声用バス（通話路）3bを接続し、これによつて通話を可能とする。

第1図は第2図の高機能端末2において行なわれる呼処理のための一実施例ブロック図であり、図中、第1図と同一のものは同一の記号で示して

あり、22aはタスク決定テーブルであり、メインメモリ22に設けられ、呼の状態、入力イベント種別及びサービスカウンタの内容により実行すべきタスクを決定するものであり、呼の状態テーブル220と、イベント種別テーブル221と、サービスカウンタテーブル222とを有しており、イベント種別テーブル221には各呼の状態に対応するイベント種別が格納され、サービスカウンタテーブル222にはイベント種別テーブル221の各イベント種別に対応して実行すべきタスク名が優先度順に格納されており、呼の状態テーブル220には各呼の状態に対応するイベント種別が格納されたイベント種別テーブル221の先頭アドレスが格納されている。22dはイベント種別レジスタ(格納部)であり、入力イベント種別が格納されるもの、22eは呼の状態レジスタであり、現呼の状態を格納しておくもの、22fはサービスカウンタであり、加算器によつてインクリメントされ、サービスカウンタ値を与えるもの、22gはGSTカウンタであり、加算器によつて

インクリメントされ、後述するGST値を与えるもの、OSはオペレーティング(制御)プログラムであり、プロセッサ21が実行するもの、APはアプリケーション(タスク)プログラムであり、実行すべきタスクプログラムを記述したものであり、第4図に示す如く、プロセッサ21のソフトウェア構成は、サービスを制御する制御プログラムOSと、サービス(タスク)を記述したサービス(タスク)プログラム群APより構成され、サービスプログラムAPは制御プログラムOSより起動される。サービスプログラムAPとしては発信処理、通常着信処理、着信拒否、キャンプオンサービス、転送サービス等が存在している。

次に第1図実施例構成の動作について第3図の呼の状態設定図及び第5図の処理フロー図により説明する。

第3図に示す如く、呼の状態としてidle, 数字受信中, 通話中, ビジートン送出中…閉塞中という第6図の800~807, 第7図の900~910で説明した従来の呼の状態の他に、仮想的にビジ

ー(busy), オール(all)といった上位概念の呼の状態を設定する。このbusyという呼の状態は電話器がフックオンされている全ての呼の状態(第6図の801~807, 第7図の901~910)に相当し、これらの呼の状態を総称した上位の呼の状態であり、allという呼の状態はbusyに含まれる呼の状態にidleという呼の状態を加えた上位の呼の状態である。

このような上位の呼の状態を設定したのは、例えばbusyという状態に含まれる個々の呼の状態において、他の端末より入力イベントとして着信要求があつた場合には、個々の呼の状態に共通に転送サービスや着信サービスというタスクを実行する様サービスが設定されていると、これらのタスクを決定する必要があることから、各個々の呼の状態に対しとの共通のタスクが決定されるようにbusyという上位の呼の状態を設定している。同様にallという状態に含まれる個々の呼の状態において、データ通知要求があつた場合には、個々の呼の状態に共通なデータ通知サービスをタスクと

して実行する様にサービスが設定されていると、個々の呼の状態に対し共通のタスクが決定されるようにallという上位の呼の状態を設定している。即ち、特定の入力イベント種別に対し各呼の状態が共通なタスクを指定する様な場合には、各呼の状態に入力イベントを設定する代りに、これら呼の状態を総称した上位の呼の状態を設定し、これによつてタスク決定を行うようにしている。

このため、タスク決定テーブル22aの呼の状態テーブル220に個々の呼の状態と共に上位の呼の状態809all, 907busy, 908allを格納しておき、これに対応してイベント種別テーブル221に共通のタスクを行うべき入力イベント種別(着信要求, データ通知要求等)を格納し、更にサービスカウンタテーブル222にこれら格納された入力イベント種別に対応して実行すべきタスク名(転送サービス, 着信サービス, データ通知サービス等)を格納している。

次に、第5図処理フロー図に従つて動作を説明する。第5図においてAは制御プログラム、Bは

サービスプログラムに対応する。

① 初期状態においてはGSTカウンタ22aのGST値は0にセットされている。この状態で入力イベントαがレジスタ22bに入力されると、プロセッサ21は制御プログラムOSによつて、GSTカウンタ22aのGST値を調べ(この場合"0")であるのでオール"all"状態を呼の状態として決定し、オール状態によつてテーブル220を索引し、テーブル221の対応する登録された入力イベント種別を得る。この入力イベント種別とレジスタ22bの入力イベント種別とを比較し、当該入力されたイベント種別がall状態でサービス(タスク)が決定できるか、即ちall状態におけるタスクを決定できる入力イベント種別かを判定する。

② この判定によつてサービスがない、即ち入力イベント種別αに対するタスクがない場合、例えば、all状態に対してはデータ通知要求の入力イベントがあつた時のみタスク(データ通知サービス)を決定するようサービスが設定されている時

定できる。

③ このbusy状態でもサービスが決定できない時には、GSTカウンタ22aをインクリメントし、"2"とし、レジスタ22cの個々の呼の状態によつてテーブル220を索引し、テーブル221の対応する入力イベント種別を求め、レジスタ22bの入力イベント種別αと比較して、サービスが決定できるかを判定する。

サービスが決定できない場合には終了する。

④ このようにして、all状態、idle状態、busy状態、個々の呼の状態で入力イベント種別αに対しサービスが決定できると判定されると、次の係にして個々のサービスプログラムが起動される。

この例では、逐次起動を可能とするため、サービスカウンタテーブル222には当該呼の状態、入力イベント種別に対し複数のタスク名が優先順位順に格納されており、これはサービスカウンタ22dのカウンタ値によつて引き出される。即ち、サービスカウンタ22dの初期値は"0"としておき、先づテーブル222の第1順位のタスク名

に、入力イベント種別がデータ通知要求以外の場合、レジスタ22cの内容により呼の状態番号を調べる。

⑤ 呼の状態番号が指定したidle状態のものに対応していればidle状態であるから、テーブル220をidle状態によつて索引し、テーブル221の対応する入力イベント種別を求め、同様にレジスタ22bの入力イベント種別αと比較する。この比較によつてサービスが決定できないと、終了する。

⑥ 逆に呼の状態番号がidle状態以外のものであればbusy状態であるので、GSTカウンタ22aをインクリメントし、"1"とし、同様にbusy状態(907)によつてテーブル220を索引し、テーブル221の対応する入力イベント種別を求め、レジスタ22bの入力イベント種別αと比較する。例えば、busy状態に対し着信要求の入力イベントがあつた時のみタスク(転送サービス等)を決定する様にサービスが設定されている時には、入力イベント種別が着信要求でない時にはサービスが決定できず、着信要求である時にはサービスが決

が読出され、当該タスク名に対応するサービスプログラムが選択され、起動処理が行なわれる。起動処理によつて第5図B列のサービスプログラムの実行に移行し、点線によつて省略した如く実行される。そして実行の結果が正常終了であるか否かを判断し、正常終了(YES)であれば、一連のサービスが実行されたのであるから、サービスカウンタ、GSTカウンタを0にリセットし、終了する。正常終了でなければ(NO)、次の優先順位のサービスを実行すべく、サービスカウンタ21aの内容を+1インクリメントし、B列のサービスプログラムの起動要因となつた入力イベントαを擬似的に発行して、図示点線の如く、A列の制御プログラムの始点に戻して再起動を行なう。この際、サービスカウンタの内容は"0"から"1"へ、"1"から"2"へと順次+1インクリメントしているため、順次優先順位の高いサービスから処理される。

例えば、busy状態で着信要求の場合に転送サービスと着信サービスとが順に登録されているとす

ると、先づ転送サービスが実行され、転送サービスが正常終了しないと蓄積サービスが実行される。

このようにして逐次起動によつてサービスが実行され、一層プログラム構成を容易にしている。

このように上位の呼の状態を設定して、特定の入力イベントに対し個々の呼の状態に共通するタスクを決定できるようにし、当該上位の呼の状態においてサービスが決定できない時に個々の呼の状態ですべてサービスを決定するようにしている。

以上の実施例では、サービスカウンタによる逐次起動の例で説明したが、これに限られることがない。また、上位の呼の状態として、all, busy について説明したが、他のものであつてもよい。更に分散処理に限らず、ネットワークコントローラ1で集中処理してもよい。

以上本発明を一実施例により説明したが、本発明は本発明の主旨に従い種々の変形が可能であり、本発明からこれらを排除するものではない。

#### 〔発明の効果〕

以上説明した様に、本発明によれば、呼の状態

と入力イベント種別に対応したタスクを実行する呼処理タスク制御方式において、該呼の状態を格納する状態記憶部と、該入力イベント種別を格納するイベント格納部と、該呼の状態と該入力イベント種別とに応じて実行すべきタスクを決定するタスク決定テーブルとを有し、該入力イベント種別に対して共通なタスクを指定する複数の呼の状態の上位の呼の状態を設定するとともに、該タスク決定テーブルに該上位の呼の状態に対応する入力イベント及びタスクを格納しておき、該イベント格納部からの入力イベントと該上位の呼の状態から該タスク決定テーブルによりタスクを決定し該上位の呼の状態に基いてタスクが決定できない時に該状態記憶部の呼の状態と該入力イベントから該タスク決定テーブルによりタスクを決定することを特徴としているので、各個々の呼の状態に対し入力イベントに応じて共通のタスクを実行する様なサービスを設定しても、個々の呼の状態に入力イベント等を個々に設定しなくてよいから、プログラムが簡素化するという効果を奏し、種々

の付加的サービスを設けることが容易となる。又変更、追加、削除も当該上位の呼の状態に対してのみ行なえばよいから、変更、追加、削除も容易であるという効果も奏し、付加的サービスを個々の加入者に応じて容易に設定でき、きめ細かなサービス機能を付与することができるという実用上優れた効果も奏する。

#### 4. 図面の簡単な説明

第1図は本発明の一実施例ブロック図、第2図は本発明が適用される分散処理形通信サービス構成図、第3図は本発明の呼の状態説明図、第4図は本発明に用いられる呼処理のソフトウェア構成図、第5図は第1図構成の一実施例処理フロー図、第6図は発信側基本接続サービス状態遷移図、第7図は着信側基本接続サービス状態遷移図である。

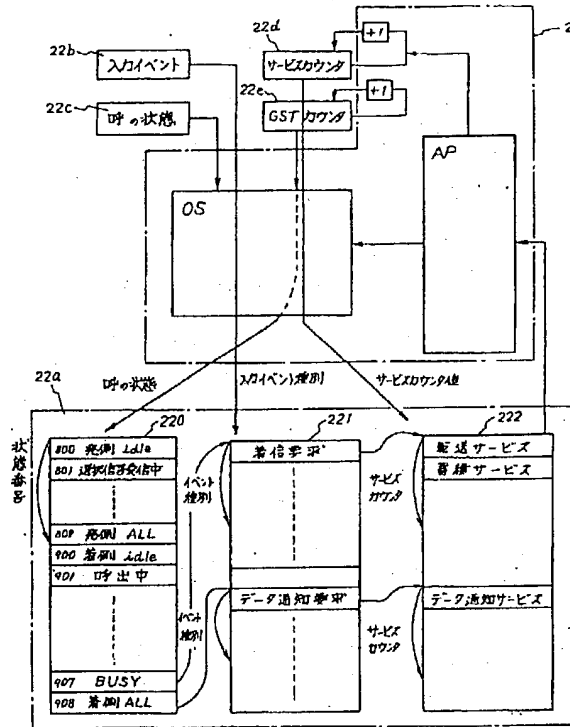
図中、1…ネットワークコントローラ、2…高機能端末、21…プロセッサ、22…ノインメモリ、22a…タスク決定テーブル、22b…入力イベントレジスタ(格納部)、22c…呼の状態

レジスタ(記憶部)、OS…制御プログラム、AP…サービスプログラム。

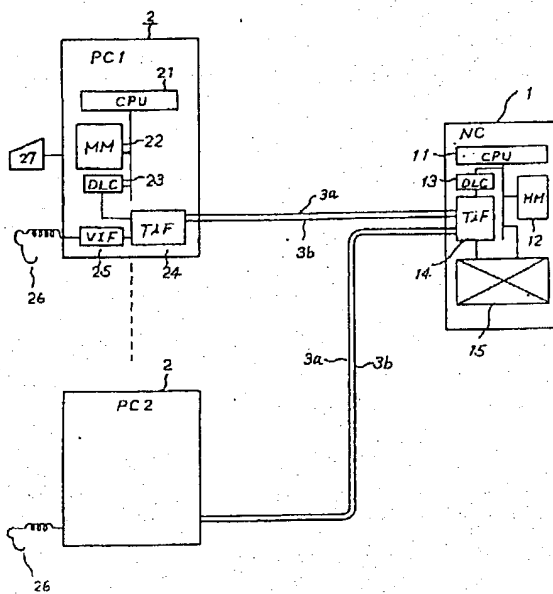
特許出願人 富士通株式会社  
代理人 弁理士 山谷 晴 榮



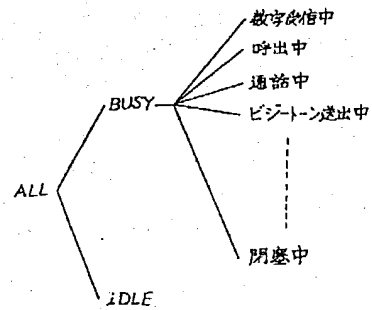
第 1 図



第 2 図



第 3 図



第 4 図

